

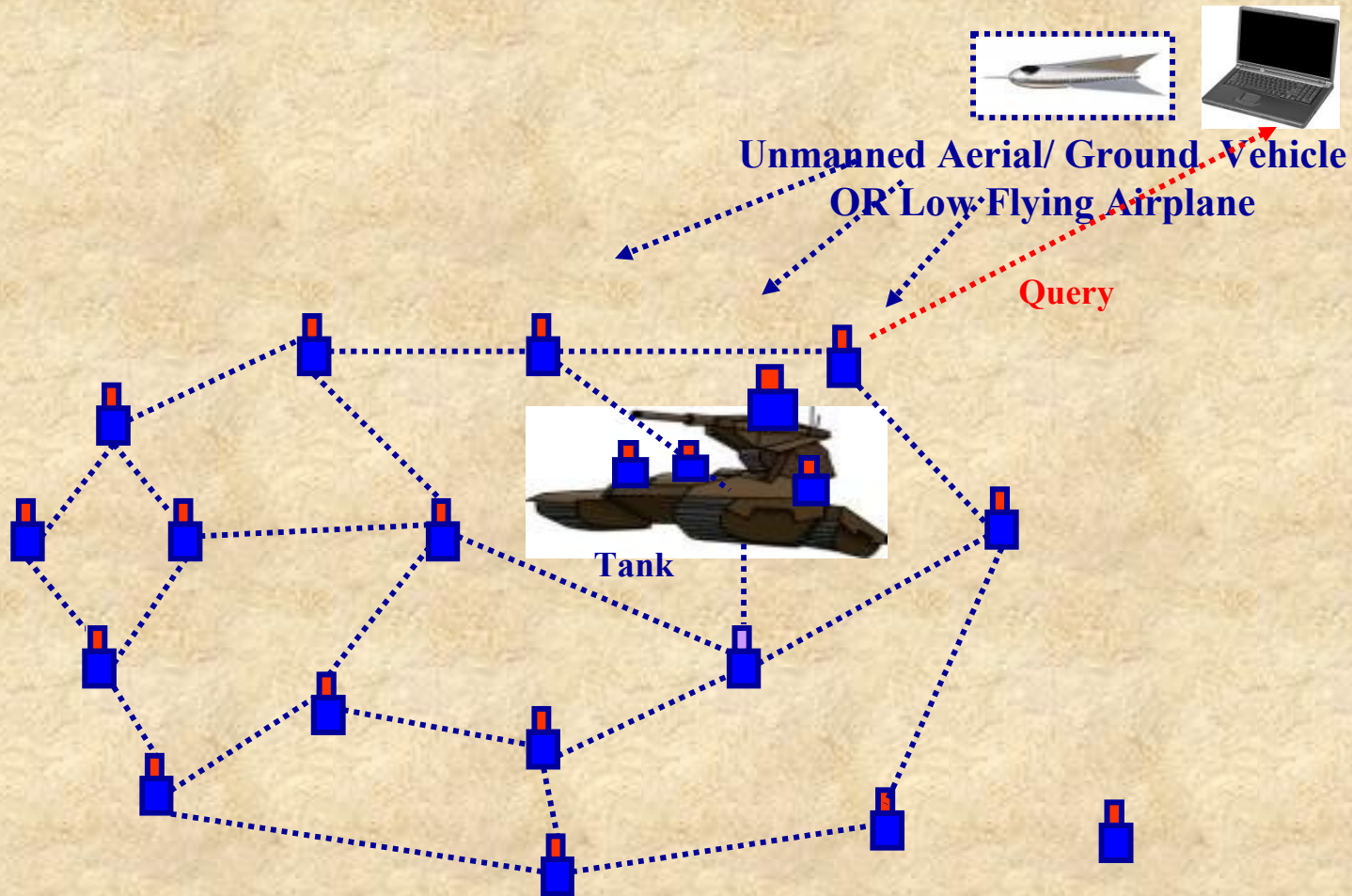


# Chapter 9: Data Retrieval in Sensor Networks

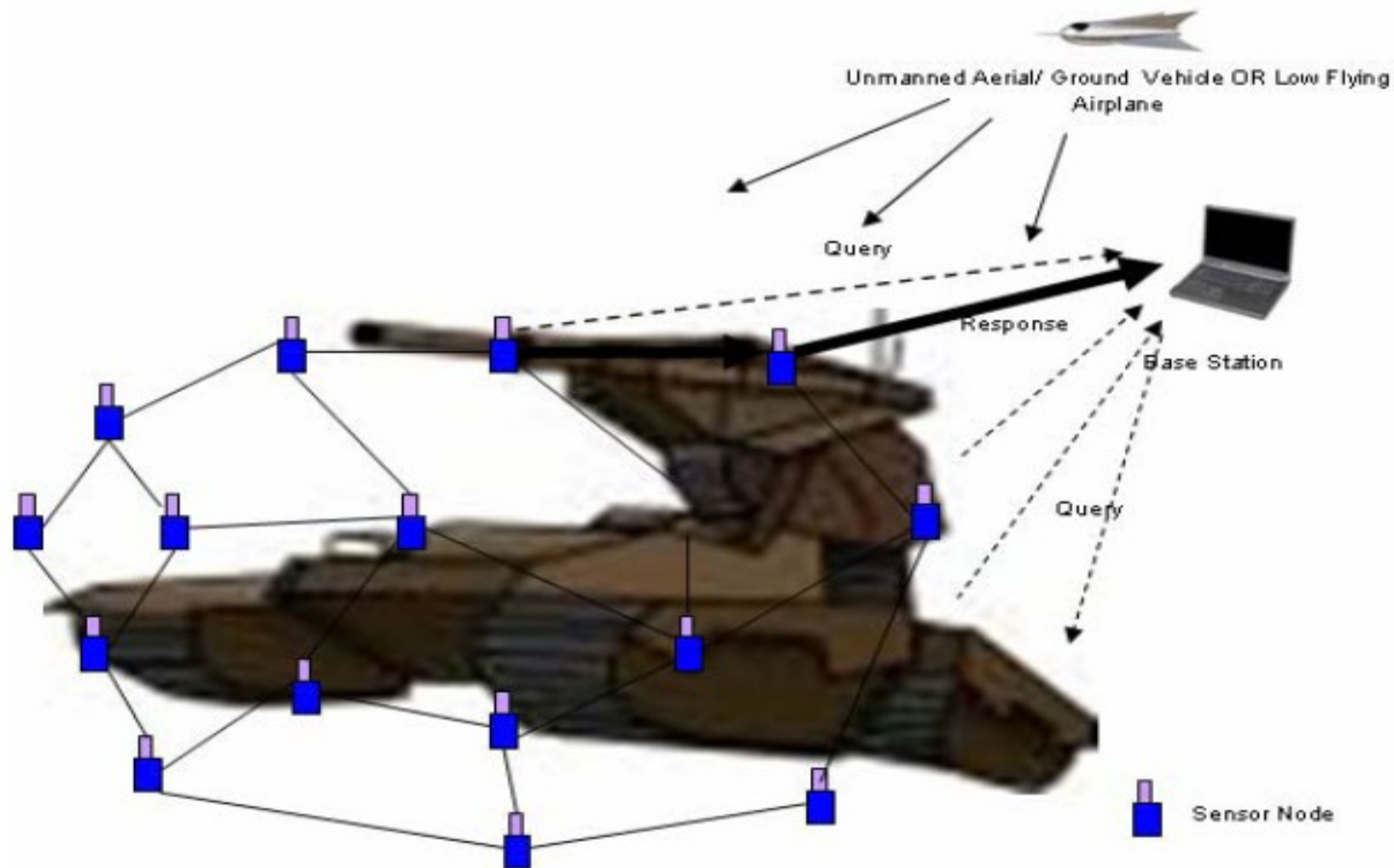
## *Table of Contents*

- **Introduction**
- **Classifications of WSNs**
  - **Architecture of Sensor Networks**
  - **Network Architecture**
  - **Physical Layer**
- **MAC Layer**
  - **Design Issues**
  - **MAC Protocols**
  - **Link Layer**
- **Routing Layer**
  - **Network Structure Based**
  - **Flat versus Hierarchical**
  - **Multipath-based Routing**
  - **Query-based Routing**
  - **Location-based Routing**
  - **Transport Layer**
- **High-Level Application Layer Support**
  - **Distributed Query Processing**
  - **Sensor Databases**
  - **Distributed Applications**
  - **In-Network Processing**
  - **Security**
- **Adapting to the Inherent Dynamic Nature of WSNs**
- **Conclusions and Future Directions**

# Introduction

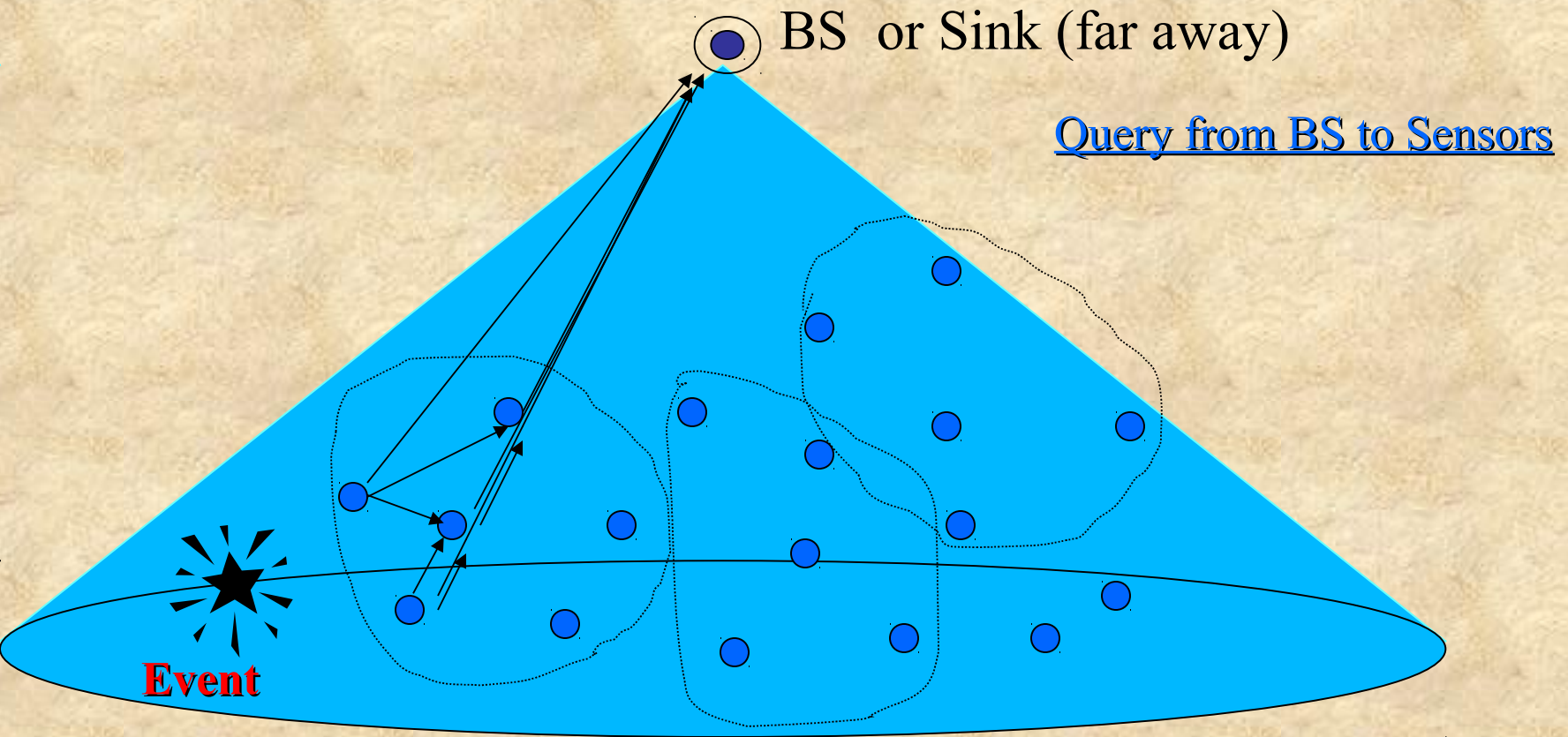


# Introduction

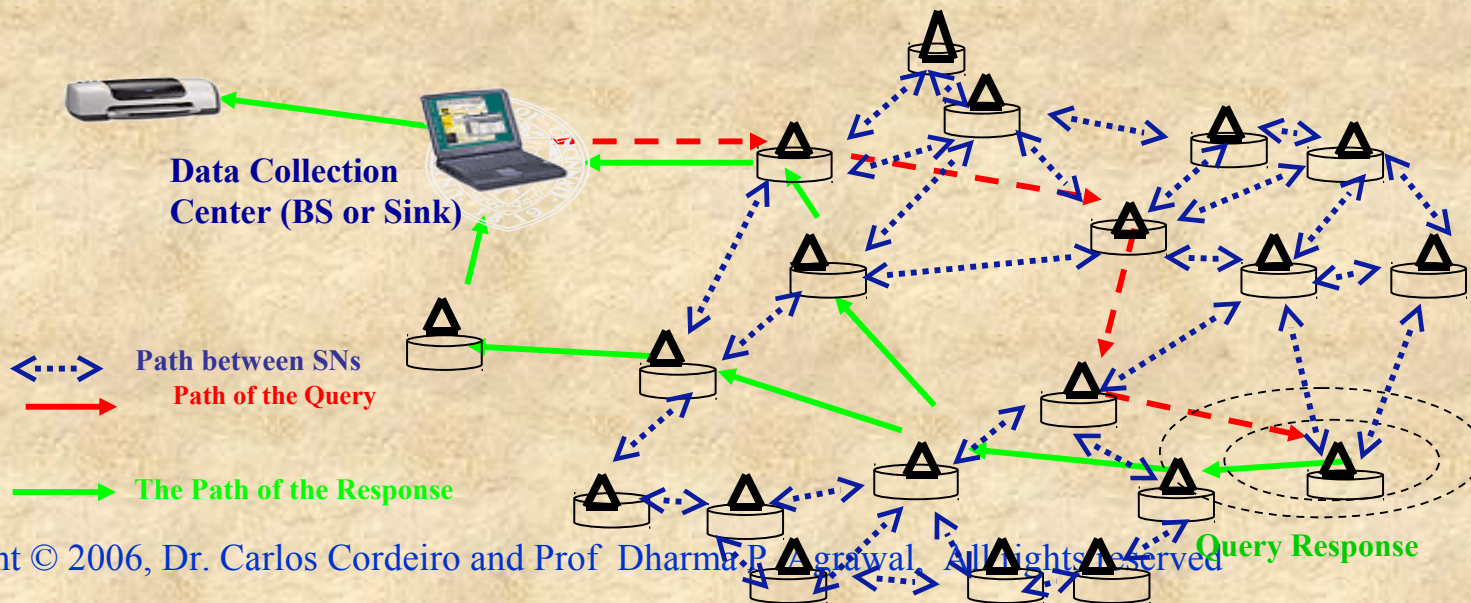
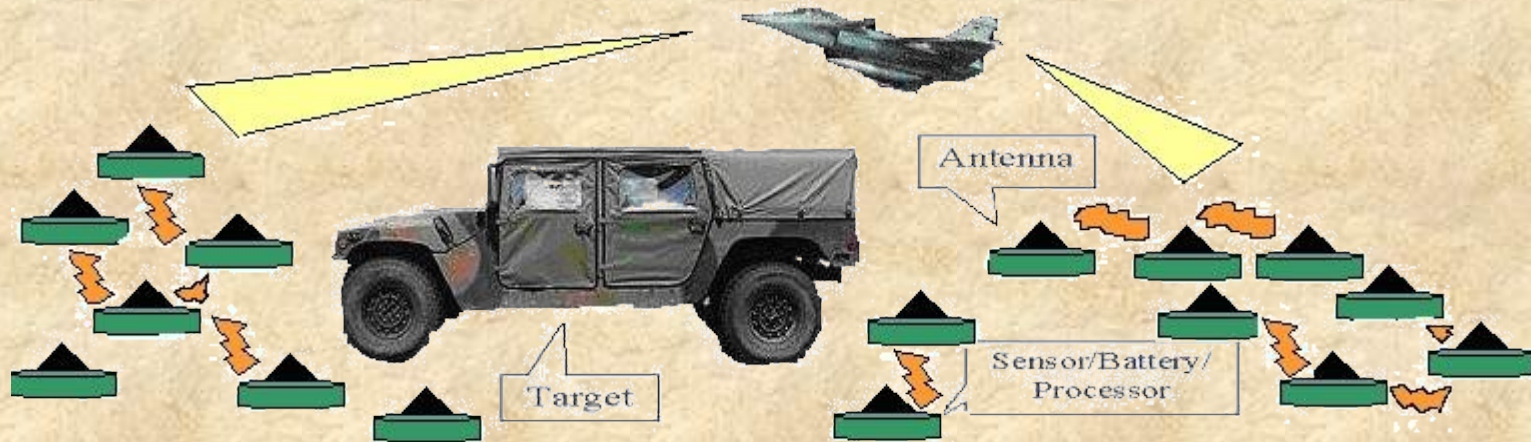




# *What is a Sensor Network?*



# Application of Wireless Sensor Networks in Defense Applications





# *Introduction*

---

- A typical Sensor Node (SN) of the network contains several transducers to measure many different physical parameters and any one could be selected under the program control at a given time
- The sensed values need to be routed by each SN to the BS either directly or via its CH in a multihop fashion due to power limitations
- In a WSN, the overall objective can be defined by the BS and this process is usually known as injection of the query by the BS
- In real-life, a low-flying airplane, an unmanned aerial or ground vehicle or a powerful laptop can act as a BS or a sink and usually have adequate source of power
- This enables the BS to transmit a query message at a very high power level so as to reach all SNs in a given area simultaneously  
Such broadcasting is to enable all SNs to start working on the request and the query could also include information about some necessary characteristics of the query





# *Introduction*

---

- If the BS has limited power to reach just few close-by SNs, then the query need to be forwarded/broadcasted to a given area of interest or possibly to the whole WSN
- The multi-hop routes are to be employed just like the response is forwarded in a multi-hop fashion
- The use of a particular type of query might depend on the application requirements
- Sometime, the query may ask for multiple parameters such as temperature, pressure, humidity, etc., and may be required to sense and transmit the values only once, or over a period of time, or use past history to gain statistical information
- Based on these, the query can be divided into three categories:
  - One time queries
  - Persistent queries
  - Historical queries



# *Classifications of WSNs*

---

- **WSNs can be classified on the basis of their mode of operation or functionality, and the type of target applications**
- **Accordingly, we classify WSNs into three types:**
  - ▣ **Proactive Networks – The nodes in this network periodically switch on their sensors and transmitters, sense the environment and transmit the data of interest and they provide a snapshot of the relevant parameters at regular intervals and are well suited for applications requiring periodic data monitoring**
  - ▣ **Reactive Networks – In this scheme, the nodes react immediately to sudden and drastic changes in the value of a sensed attribute and as such, these are well suited for time critical applications**
  - ▣ **Hybrid Networks – This is a combination of both proactive and reactive networks where sensor nodes not only send data periodically, but also respond to sudden changes in attribute values**





# *Architecture of Sensor Networks*

---

**The typical hardware platform of a wireless sensor node will consist of:**

- **A simple embedded microcontrollers, such as the Atmel or the Texas Instruments MSP 430**
- **Currently used radio transceivers include the RFM TR1001 or Infineon or Chipcon devices**
- **Typically, ASK or FSK is used, while the Berkeley PicoNodes employ OOK modulation**
- **Radio concepts like ultra-wideband are in an advanced stage**
- **Batteries provide the required energy as an important concern is battery management and whether and how energy scavenging can be done to recharge batteries in the field**
- **The operating system and the run-time environment is a hotly debated issue in the literature**
- **On one hand, minimal memory footprint and execution overhead are required while on the other, flexible means of combining protocol building blocks are necessary, as meta information has to be used in many places in a protocol stack**

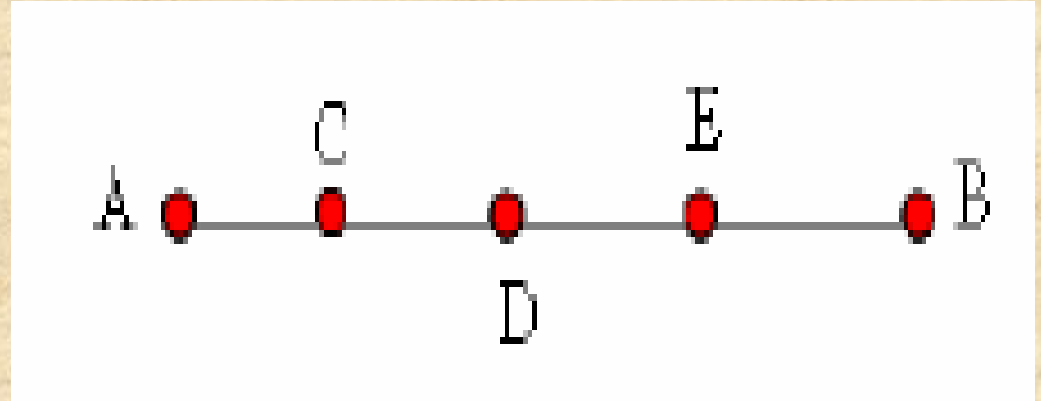
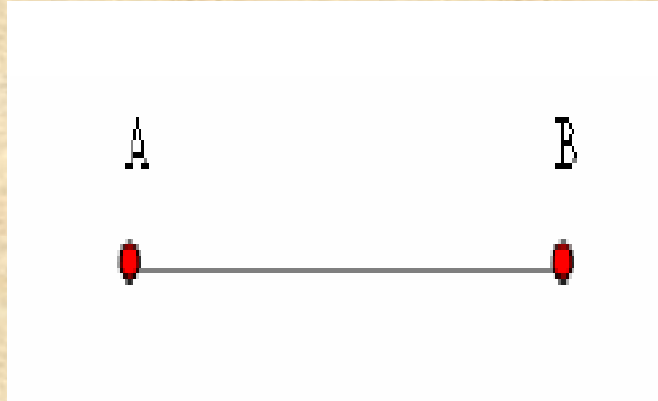


# *Network Architecture*

---

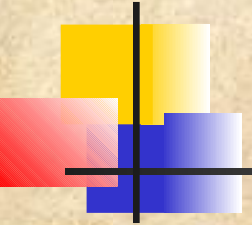
- WSN architecture need to cover a desired area both for sensing coverage and communication connectivity point of view
- Therefore, density of the WSN network is critical for the effective use of the WSN
- There is no well-defined measure of life-time of a WSN and some assume either the failure of a single sensor running out of battery power as life-time of the network
- Perhaps a better definition is if certain percentage of sensors stops working, may define the life-time as the network continues to operate
- The percentage failure may depend on the nature of application and as long as the area is adequately covered by the operating sensors, a WSN may be considered operational
- The SNs are yet to become inexpensive to be deploying with some degree of redundancy

# Network Architecture



- There is an optimal distance between two sensors that would maximize the sensor lifetime
- So, if the density of sensors is high, then some of the sensors can be put into sleep mode to have close to optimal distance between the sensors
- Very little work has been done on protocols that suits well to the needs of WSNs
- With respect to the radio transmission, the main question is how to transmit as energy efficiently as possible, taking into account all related costs (possible retransmissions, overhead, and so on)





# *MAC Layer*

---

- The MAC and the routing layers are the most active research areas in WSNs
- There are two types of schemes available to allocate a single broadcast channel among competing nodes: **Static Channel Allocation** and **Dynamic Channel Allocation**
  - **Static Channel Allocation:** In this category of protocols, if there are N SNs, the bandwidth is divided into N equal portions in frequency (FDMA), in time (TDMA), in code (CDMA), in space (SDMA) or in schemes such as OFDM or ultra-wideband and since each SN is assigned a private portion, there is no or minimal interference amongst multiple SNs
  - **Dynamic Channel Allocation:** In this category of protocols, there is no fixed assignment of bandwidth
- When the number of active SNs changes dynamically and data becomes bursty at arbitrary SNs, it is most advisable to use dynamic channel allocation scheme
- These are contention-based schemes, where SNs contend for the channel when they have data while minimizing collisions with other SNs' transmissions



# *MAC Protocols*

---

- WSNs are designed to operate for long time as it is rather impractical to replenish the batteries
- However, nodes are in idle state for most time when no sensing occurs
- Measurements have shown that a typical radio consumes the similar level of energy in idle mode as in receiving mode
- Therefore, it is important that nodes are able to operate in low duty cycles

## *The Sensor-MAC*

- The Sensor-MAC (S-MAC) protocol explores design trade-offs for energy-conservation in the MAC layer
- It reduces the radio energy consumption from the following sources: collision, control overhead, overhearing unnecessary traffic, and idle listening
- The basic scheme of S-MAC is to put all SNs into a low-duty-cycle mode – listen and sleep periodically
- When SNs are listening, they follow a contention rule to access the medium, which is similar to the IEEE 802.11 DCF





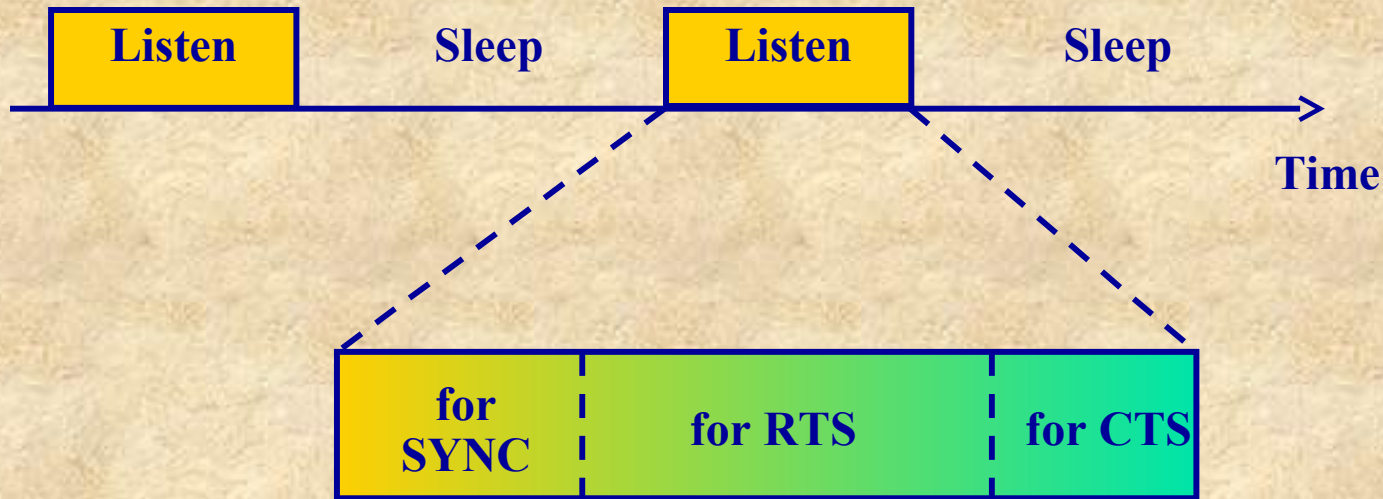
# *Sensor-MAC*

---

- In S-MAC, SNs exchange and coordinate on their sleep schedules rather than randomly sleep on their own
- Before each SN starts the periodic sleep, it needs to choose a schedule and broadcast it to its neighbors
- To prevent long-term clock drift, each SN periodically broadcasts its schedule as the SYNC packet
- To reduce control overhead and simplify broadcasting, S-MAC encourages neighboring SNs to choose the same schedule, but it is not a requirement
- A SN first listens for a fixed amount of time, which is at least the period for sending a SYNC packet
- If it receives a SYNC packet from any neighbor, it will follow that schedule by setting its own schedule to be the same
- Otherwise, the SN will choose an independent schedule after the initial listening period



# *Sensor-MAC*



- Figure depicts the low-duty-cycle operation of each SN
- The listen interval is divided into two parts for both SYNC and data packets
- There is a contention window for randomized carrier sense time before sending each SYNC or data (RTS or broadcast) packet



# *SMACS*

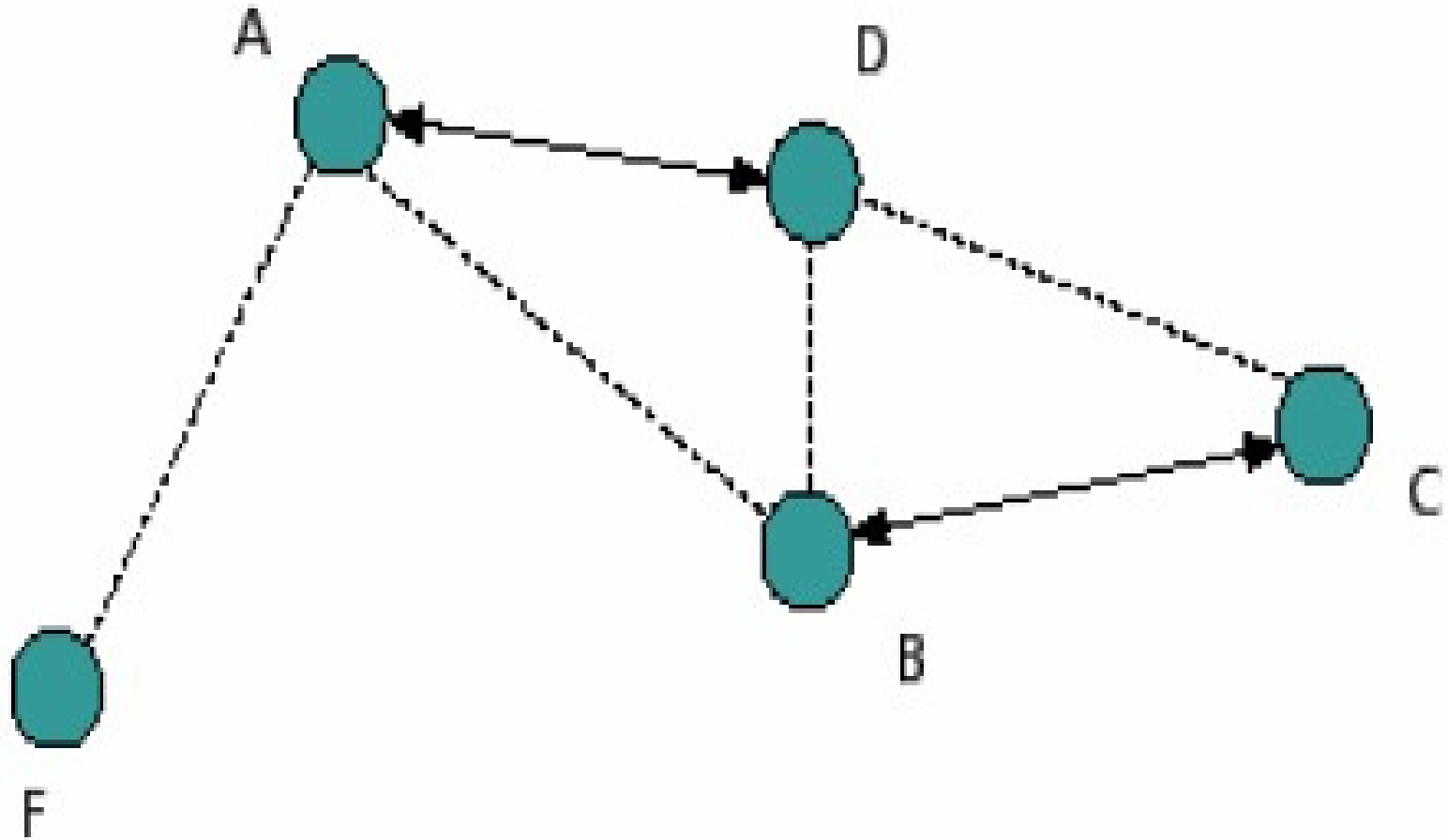
---

- The SMACS is an infrastructure-building protocol that forms a flat topology (as opposed to a cluster hierarchy) for sensor networks
- SMACS is a distributed protocol which enables a collection of SNs to discover their neighbors and establish transmission/reception schedules for communicating with them without the need for any local or global master nodes
- In order to achieve this ease of formation, SMACS combines the neighbor discovery and channel assignment phases
- SMACS assigns a channel to a link immediately after the link's existence is discovered
- This way, links begin to form concurrently throughout the network
- By the time all nodes hear all their neighbors, they would have formed a connected network
- In a connected network, there exists at least one multihop path between any two distinct nodes

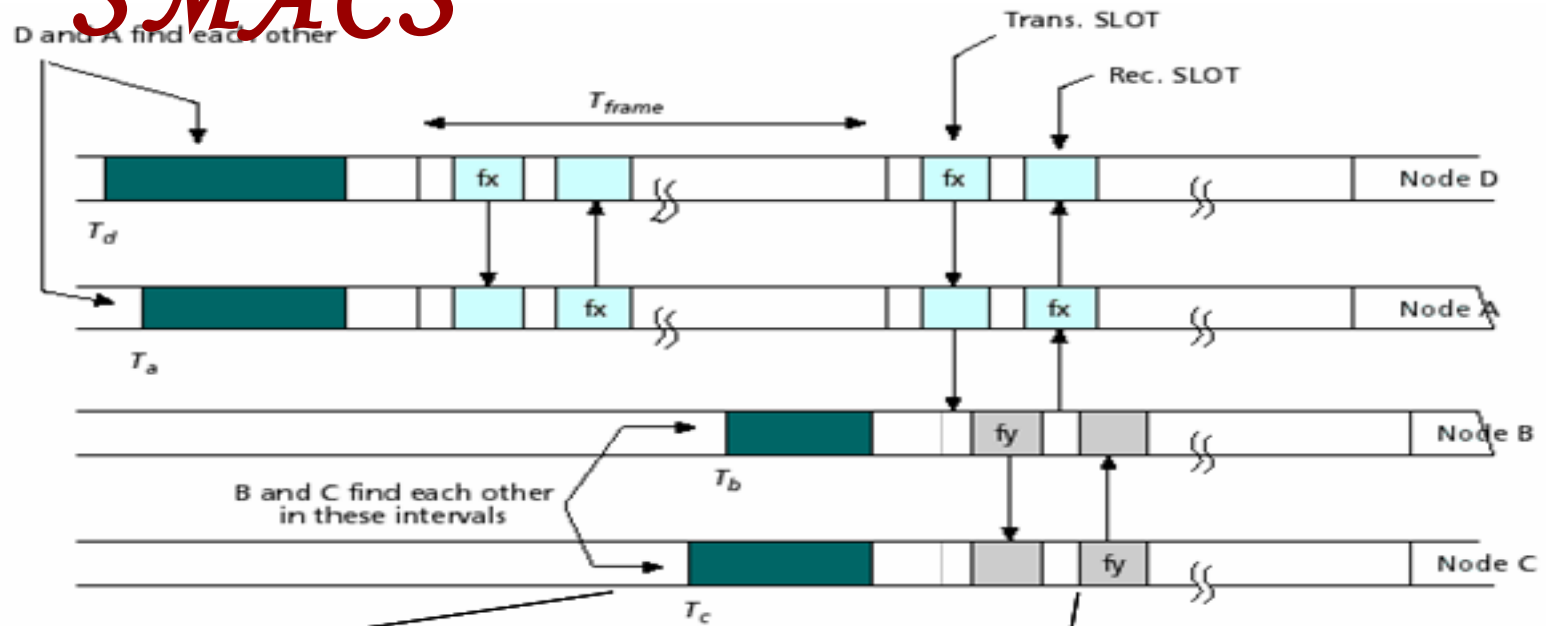
- Since only partial information about radio connectivity in the vicinity of a SN is used to assign time intervals to links, there is a potential for time collisions with slots assigned to adjacent links whose existence is not known at the time of channel assignment
- To reduce the likelihood of collisions, each link is required to operate on a different frequency
- This frequency band is chosen at random from a large pool of possible choices when the links are formed
- This idea is described in Figure 9.6(a) for the topology of Figure 9.5
- Here, nodes A and D wake up at times  $T_a$  and  $T_d$
- After they find each other, they agree to transmit and receive during a pair of fixed time slots
- This transmission/reception pattern is repeated periodically every  $T_{\text{frame}}$
- Nodes B and C, in turn, wake up later at times  $T_b$  and  $T_c$ , respectively



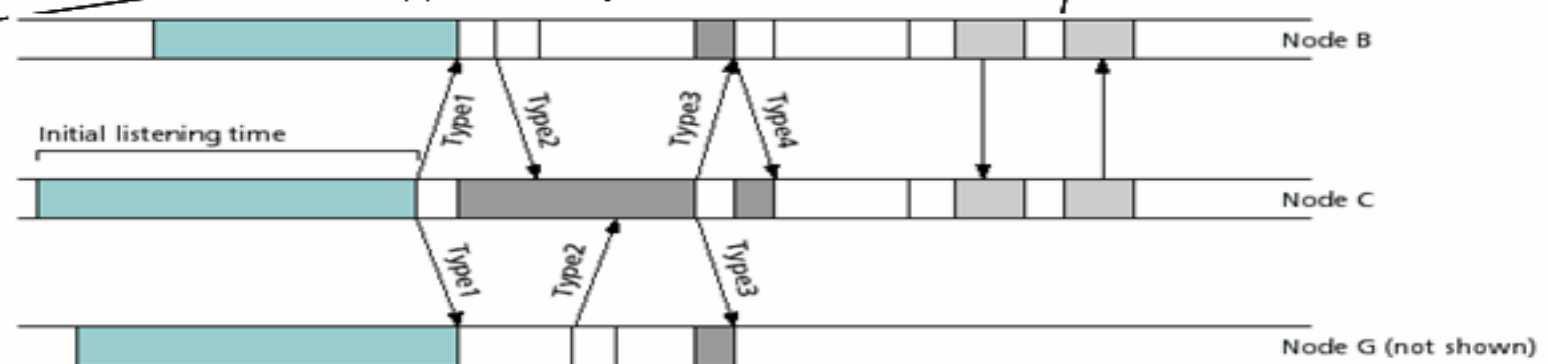
# *Network Topology*



# Node Discovery Phase in SMACS



(a) – Nonsynchronous communications



(b) Node discovery phase



# *SMACS*

---

- The method by which SNs find each other and the scheme by which time slots and operating frequencies are determined constitute an important part of SMACS
- To illustrate this scheme, consider nodes B, C, and D shown in Figure 9.6(b)
- These nodes are engaged in the process of finding neighbors and wake up at random times
- Upon waking up, each node listens to the channel on a fixed frequency band for some random time duration
- A node decides to transmit an invitation by the end of this initial listening time if it has not heard any invitations from other nodes
- This is what happens to node C, which broadcasts an invitation or TYPE1 message
- Nodes B and D hear this TYPE1 message and broadcasts a response, or TYPE2, message addressed to node C during a random time following reception of TYPE1





# ***EAR (Eaves-drop-And-Register)***

---

- Mobility can be introduced into a WSN as extensions to the stationary sensor network
- Mobile connections are very useful to a WSN and can arise in many scenarios where either energy or bandwidth is a major concern
- Where there is the constraint of limited power consumption, small, low bit rate data packets can be exchanged to relay data to and from the network whenever necessary
- At the same time, it cannot be assumed that each mobile node is aware of the global network state and/or node positions
- Similarly, a mobile node may not be able to complete its task (data collection, network instruction, information extraction) while remaining motionless
- Thus, the EAR protocol attempts to offer continuous service to these mobile nodes under both mobile and stationary constraints
- EAR is a low-power protocol that allows for operations to continue within the stationary network while intervening at desired moments for information exchange

**The EAR algorithm employs the following four primary messages:**

- 1. Broadcast Invite (BI) – This is used by the stationary node to invite other nodes to join and if multiple BIs are received, the mobile node continues to register every stationary node encountered, until its registry becomes full**
- 2. Mobile Invite (MI) – This message is sent by the mobile node in response to the BI message from the stationary node for establishing connection**
- 3. Mobile Response (MR) – This is sent by the stationary node in response to a MI message, and indicates the acceptance of the MI request which causes the stationary node to select slots along the TDMA frame for communication and the stationary node will enter the mobile node in its own registry**
- 4. Mobile Disconnect (MD) – With this message, the mobile node informs the stationary node of a disconnection and for energy saving purposes, no response is needed from the stationary node**





# *The STEM*

---

- The Sparse Topology and Energy Management (STEM) protocol is based on the assumption that most of the time the sensor network is only sensing the environment, waiting for an event to happen
- In other words, STEM may be seen as better suitable for reactive sensor networks where the network is in the monitoring state for vast majority of time
- One example of such an application is a sensor network designed to detect fires in a forest
- These networks have to remain operational for months or years, but sensing only on the occurrence of a forest fire
- Clearly, although it is desirable that the transfer state be energy-efficient, it may be more important that the monitoring state be ultra-low-power as the network resides in this state for most of the time
- This observation holds true for many other applications as well





# *The STEM*

---

- The idea behind STEM is to turn on only a node's sensors and some preprocessing circuitry during monitoring states
- Whenever a possible event is detected, the main processor is woken up to analyze the sensed data in detail and forward it to the data sink
- However, the radio of the next hop in the path to the data sink is still turned off, if it did not detect the same event
- STEM solves this problem by having each node to periodically turn on its radio for a short time to listen if someone else wants to communicate with it
- The node that wants to communicate, i.e. , the initiator SN, sends out a beacon with the ID of the node it is trying to wake up, i.e. , the target SN
- This can be viewed as a procedure by which the initiator SN attempts to activate the link between itself and the target SN
- As soon as the target SN receives this beacon, it responds back to the initiator node and both keep their radio on at this point



# *The STEM*

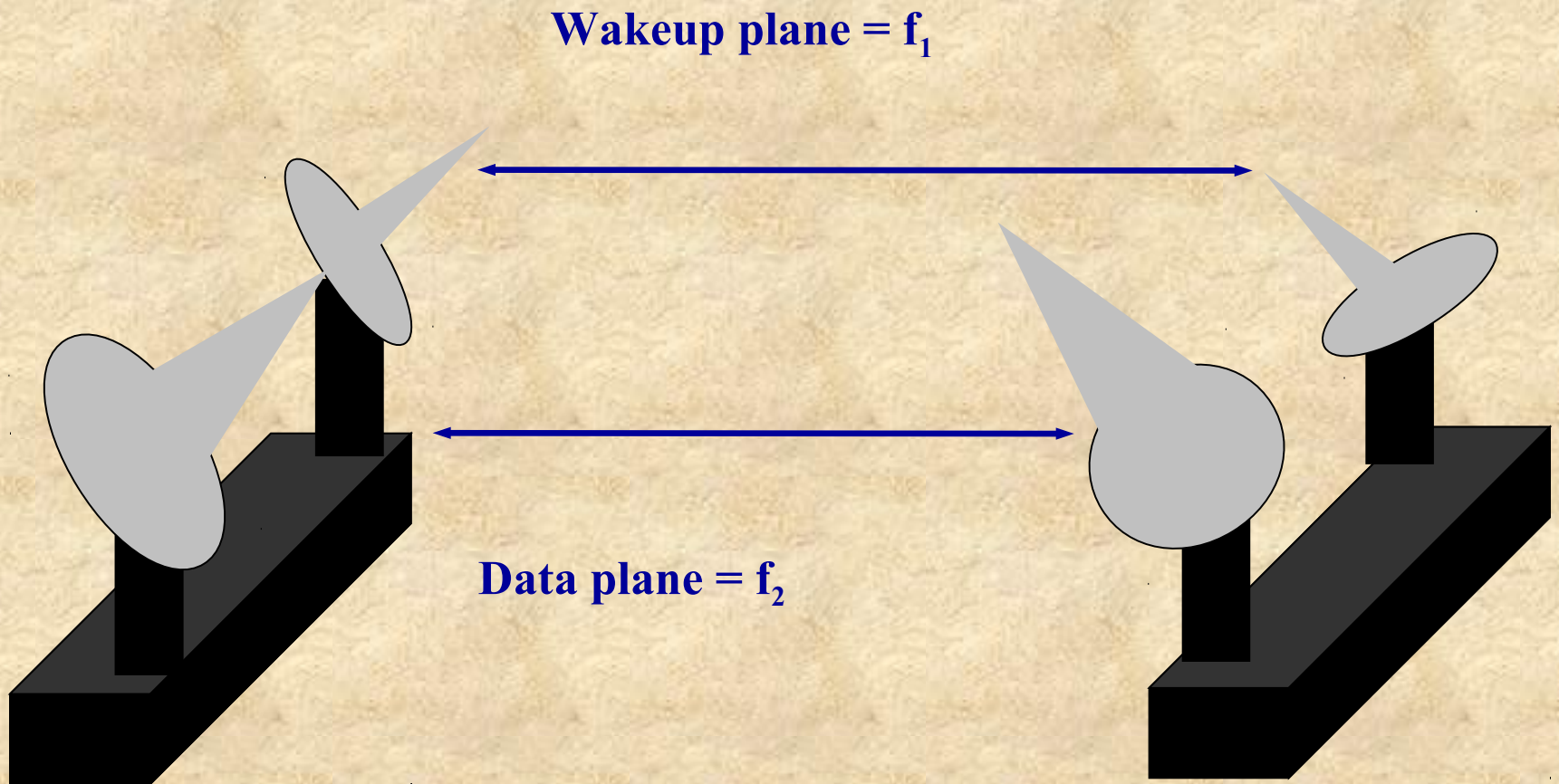
---

- If the packet needs to be relayed further, the target SN will become the initiator node for the next hop and the process is repeated
- Once both the nodes that make up a link have their radio on, the link is active and can be used for subsequent packets
- However, the actual data transmissions may still interfere with the wakeup protocol
- To overcome this problem, STEM proposes the wakeup protocol and the data transfer to employ different frequency bands as depicted in Figure 9.7
- In addition, separate radios would be needed in each of these bands
- In Figure 9.7 we see that the wakeup messages are transmitted by the radio operating in frequency band  $f_1$
- STEM refers to these communications as occurring in the *wakeup plane*
- Once the initiator SN has successfully notified the target SN, both SNs turn on their radio that operates in frequency band  $f_2$
- The actual data packets are transmitted in this band, called the *data plane*



# *The Stem*

---







# *Routing Layer*

---

- **Routing in sensor networks is usually multi-hop**
- **The goal is to send the data from source node(s) to a known destination node**
- **The destination node or the sink node is known and addressed by means of its location**
- **A BS may be fixed or mobile, and is capable of connecting the sensor network to an existing infrastructure where the user can have access to the collected data**
- **The task of finding and maintaining routes in WSNs is nontrivial since energy restrictions and sudden changes in node status (e.g., failure) cause frequent unpredictable topological changes**
- **Thus, the main objective of routing techniques is to minimize the energy consumption in order to prolong WSN lifetime**
- **To achieve this objective, routing protocols proposed in the literature employ some well-known routing techniques as well as tactics special to WSNs**
- **To preserve energy, strategies like data aggregation and in-network processing, clustering, different node role assignment, and data-centric methods are employed**



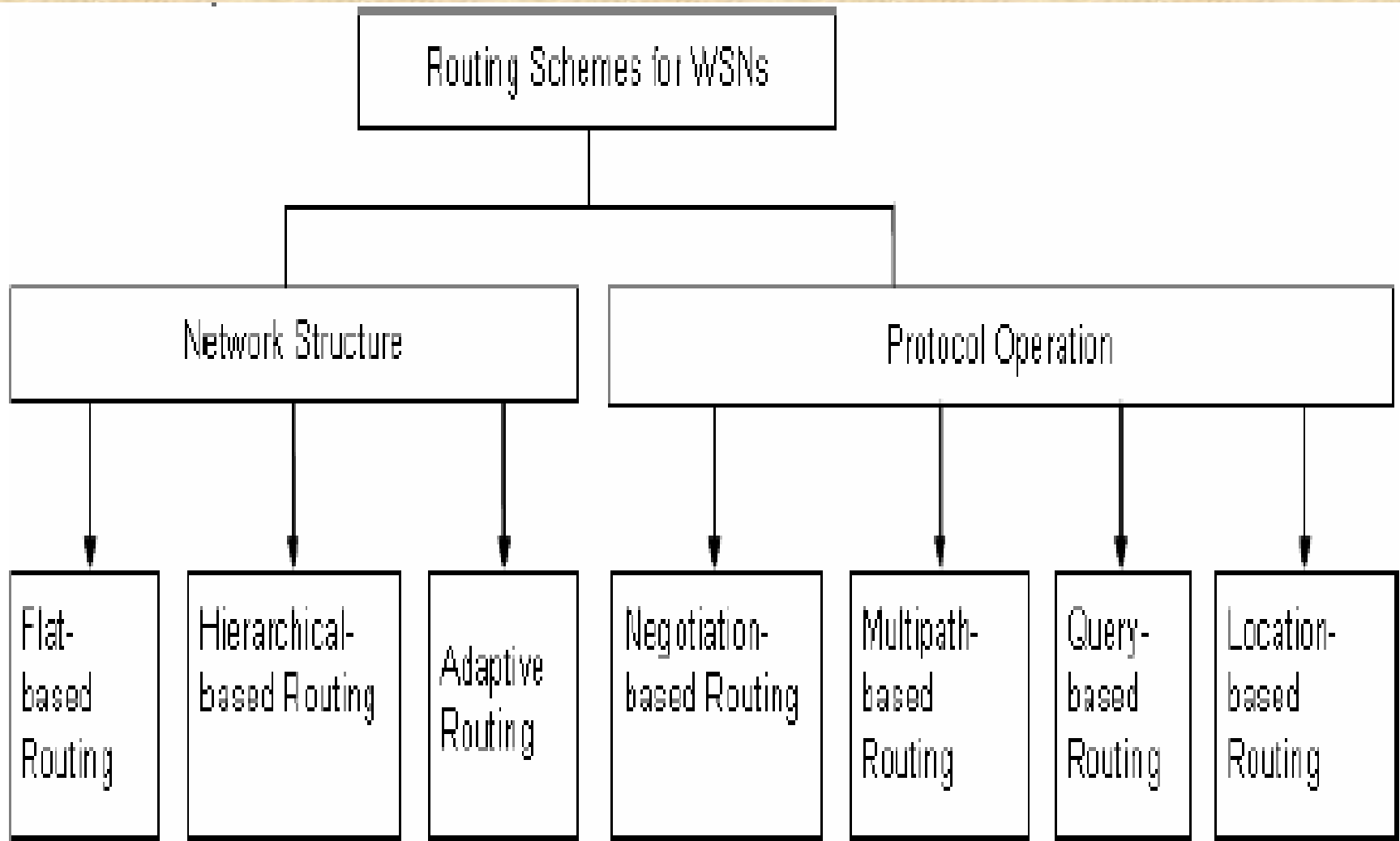
# *Routing Layer*

---

- In sensor networks, conservation of energy is considered relatively more important than quality of data sent
- Therefore, energy-aware routing protocols need to satisfy this requirement
- Routing protocols for WSNs have been extensively studied in the last few years
- Routing protocols for WSNs can be broadly classified into flat-based, hierarchical-based, and adaptive, depending on the network structure
- In flat-based routing, all nodes are assigned equal role
- In hierarchical-based routing, however, nodes play different roles and certain nodes, called cluster heads (CHs), are given more responsibility
- In adaptive routing, certain system parameters are controlled in order to adapt to the current network conditions and available energy levels
- Furthermore, these protocols can be classified into multipath-based, query-based, negotiation-based, or location-based routing techniques



# *Routing Layer*







# *Network Structure Based*

---

- In this class of routing protocols, the network structure is one of the determinant factors
- In addition, the network structure can be further subdivided into flat, hierarchical and adaptive depending upon its organization

## **Flat Routing**

- In flat routing based protocols, all nodes play the same role and we present the most prominent protocols falling in this category

## **Directed Diffusion**

- Directed Diffusion is a data aggregation and dissemination paradigm for sensor networks
- It is a data-centric (DC) and application-aware approach in the sense that all data generated by sensor nodes is named by attribute-value pairs
- Directed Diffusion is very useful for applications requiring dissemination and processing of queries
- The main idea of the DC paradigm is to combine the data coming from different sources en-route (in-network aggregation) by eliminating redundancy, minimizing the number of transmissions; thus saving network energy and prolonging its lifetime



# *Data Centric Routing and Directed Diffusion*

---

- Unlike traditional end-to-end routing, DC routing finds routes from multiple sources to a single destination (BS) that allows in-network consolidation of redundant data
- In Directed Diffusion, sensors measure events and create gradients of information in their respective neighborhoods
- The BS requests data by broadcasting interests, which describes a task to be done by the network
- Interest diffuses through the network hop-by-hop, and is broadcast by each node to its neighbors
- As the interest is propagated throughout the network, gradients are setup to draw data satisfying the query towards the requesting node
- Each SN that receives the interest setup a gradient toward the SNs from which it receives the interest
- This process continues until gradients are setup from the sources back to the BS





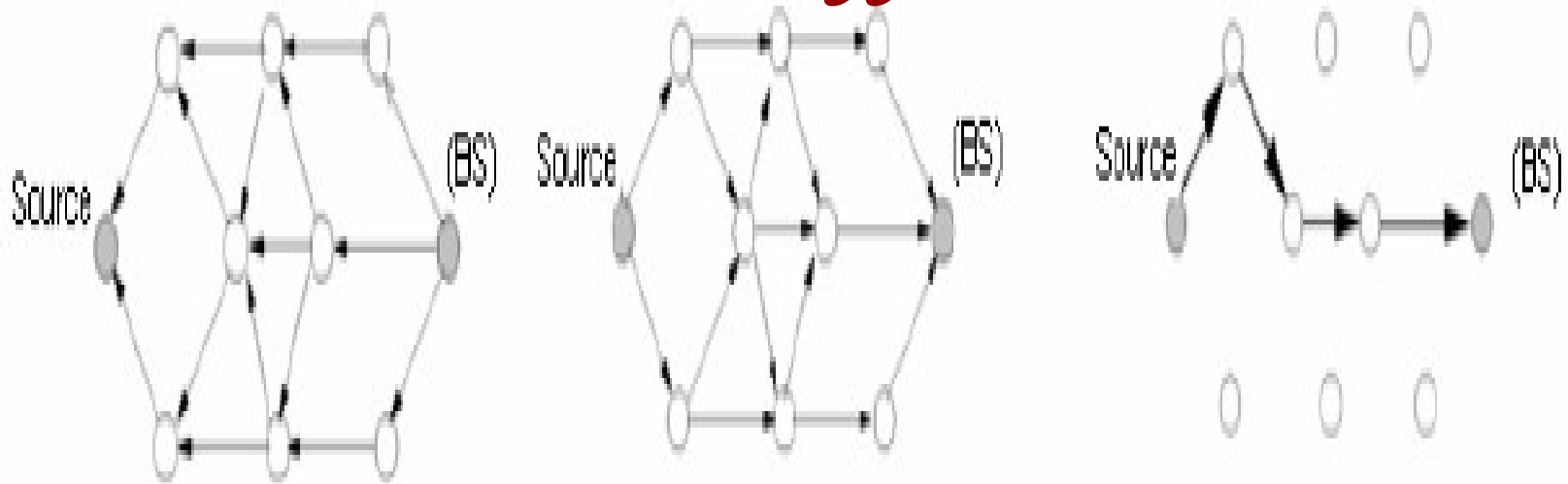
# *Data Centric Routing and Directed Diffusion*

---

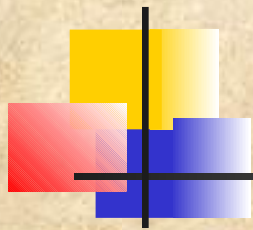
- The strength of the gradient may be different towards different neighbors, resulting in variable amounts of information flow
- At this point, loops are not checked, but are removed at a later stage
- Figure 9.9 depicts an example of the operation of directed diffusion
- Figure 9.9(a) presents the propagation of interests, Figure 9.9(b) shows the gradients construction, and Figure 9.9(c) depicts the data dissemination
- When interests fit gradients, paths of information flow are formed from multiple paths, and the best paths are reinforced so as to prevent further flooding according to a local rule
- In order to reduce communication costs, data is aggregated on the way  
The BS periodically refreshes and re-sends the interest when it starts to receive data from the source(s)
- This retransmission of interests is needed because the medium is inherently unreliable



# *Data Centric Routing and Directed Diffusion*



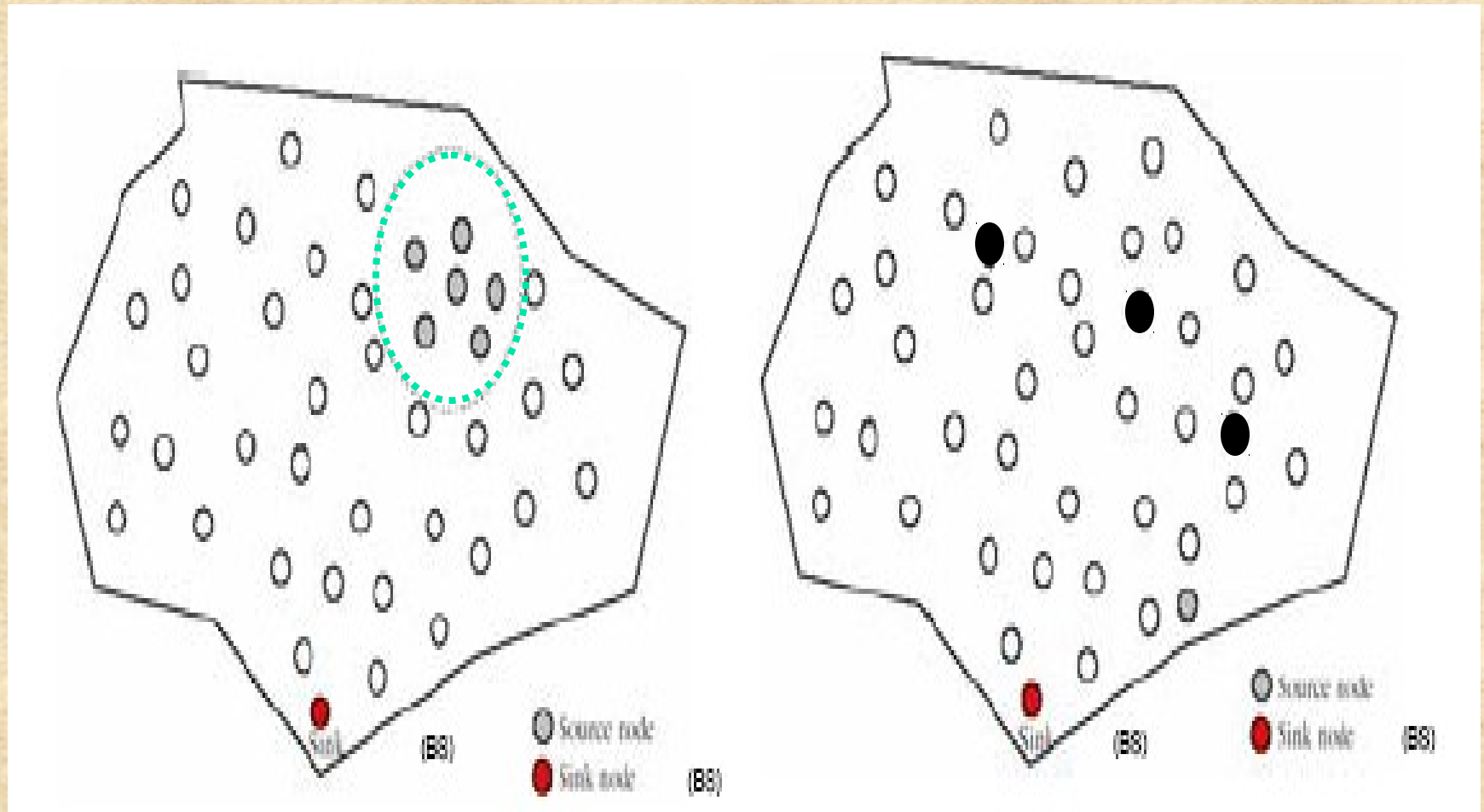
- Sensor nodes in a directed diffusion-based network are application-aware, which enables diffusion to achieve energy savings by choosing empirically good paths and by caching and processing data in the network
- An application of directed diffusion is to spontaneously propagate an important event to regions of the sensor network
- Such type of information retrieval is well suited for persistent queries where requesting nodes expect data that satisfy a query for a period of time



# *Data Centric Routing and Directed Diffusion*

- The performance of data aggregation methods employed by the directed diffusion paradigm is affected by the location of the source nodes in the network, the number of sources, and the network topology
- In order to investigate these factors, two models of source placement shown in Figure 9.10 have been investigated
- These models are called the **event radius (ER)** model, and the **random sources (RS)** model
- In the **ER model**, a single point in the network area is defined as the location of an event
- For example, this may correspond to a vehicle or some other phenomenon being tracked by the sensor nodes
- All nodes within a distance  $S$  (called the sensing range) of this event that are not sinks, are considered to be data sources
- In the **RS model**,  $K$  nodes that are not sinks are randomly selected to be sources
- Unlike the ER model, in the RS model the sources are not necessarily close to each other

# *Network Structure Based*



**Event Radius (ER) model**

**Random source (RS) model**





# *Sequential Assignment Routing*

## *(SAR)*

- The routing scheme in SAR depends on three factors: **energy resources, QoS on each path, and the priority level of each packet**
- To avoid single route failure, a multi-path approach coupled with a localized path restoration scheme is employed
- To create multiple paths from a source node, a tree rooted at the source node to the destination nodes (i.e., the set of BSs) is constructed
- The paths of the tree are defined by avoiding nodes with low energy or QoS guarantees
- At the end of this process, each sensor node is a part of multi-path tree
- For each SN, two metrics are associated with each path: **delay** (which is an additive QoS metric); and **energy usage** for routing on that path
- The energy is measured with respect to how many packets will traverse that path
- SAR calculates a weighted QoS metric as the product of the additive QoS metric and a weight coefficient associated with the priority level of the packet
- The goal of SAR is to minimize the average weighted QoS metric for the network



# *Minimum Cost Forwarding Algorithm*

---

- The minimum cost forwarding algorithm (MCFA) exploits the fact that the direction of routing is always known, that is, towards fixed and predetermined external BS
- Therefore, a SN need not have a unique ID nor maintain a routing table
- Instead, each node maintains the least cost estimate from itself to the BS
- Each message forwarded by the SN is broadcast to its neighbors
- When a node receives the message, it checks if it is on the least cost path between the source SN and the BS
- If so, it re-broadcasts the message to its neighbors
- This process repeats until the BS is reached
- In MCFA, each sensor node should know the least cost path estimate from itself to the BS





# *Coherent and Non-Coherent Processing*

---

- Data processing is a major component in the operation of any WSN
- In general, sensor nodes cooperate with each other in processing different data flooded throughout the network
- Two examples of data processing techniques are **coherent** and **non-coherent data processing-based routing**
- In non-coherent data processing routing, nodes locally process the raw data before being sent to other nodes for further processing
- The nodes that perform further processing are called the aggregators
- In coherent routing, the data is forwarded to aggregators after minimum processing of time stamping and duplicate suppression
- To perform energy-efficient routing, normally coherent processing is selected





# *Energy Aware Routing*

---

- This protocol is similar to directed diffusion (discussed earlier) with the difference that it maintains a set of paths instead of maintaining or enforcing one optimal path
- These paths are maintained and chosen by means of a certain probability, which depends on how low the energy can be conserved for each path
- By selecting different routes at different times, the energy of any single route will not deplete so quickly
- The protocol initiates a connection through localized flooding, which is used to discover all routes between source/destination pair and their costs; thus building up the routing tables
- Next, the high-cost paths are discarded and a forwarding table is constructed by choosing neighboring nodes inversely proportional to their cost
- Then, data is sent to the destination using the forwarding table with a probability that is inversely proportional to the node cost
- Finally, to keep the various paths alive, localized flooding is carried out by the destination node

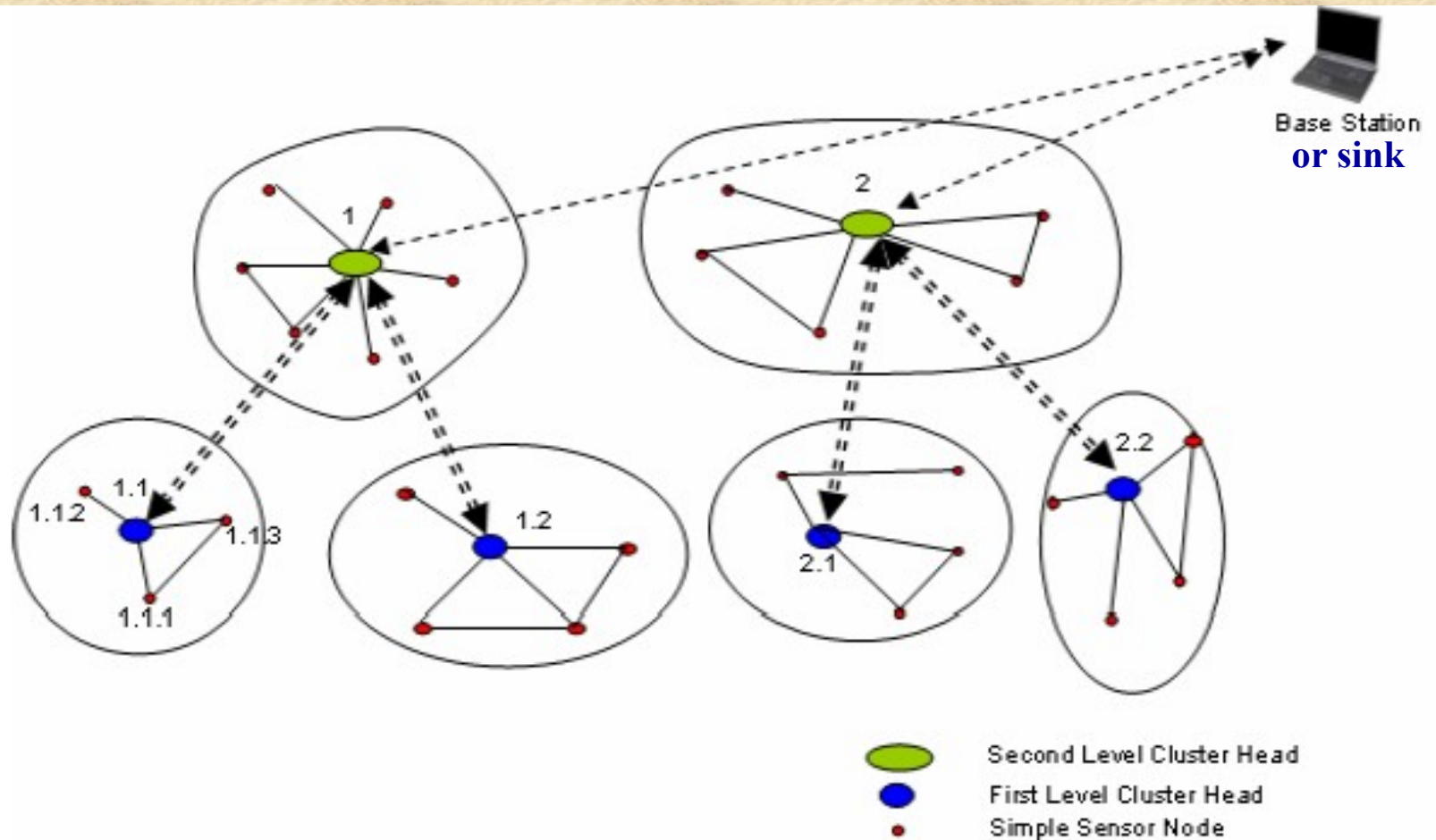


# *Hierarchical Routing*

---

- Hierarchical, or cluster-based routing has its roots in wired networks, where the main goals are to achieve scalable and efficient communication
- As such, the concept of hierarchical routing has also been employed in WSN to perform energy-efficient routing
- In a hierarchical architecture, higher energy nodes (usually called **cluster heads**) can be used to process and send the accumulated information while low energy nodes can be used to sense in the neighborhood of the target and pass on to the **CH**
- In these cluster-based architectures creation of clusters and appropriate assignment of special tasks to CHs can contribute to overall system scalability, lifetime, and energy efficiency
- An example of a general hierarchical clustering scheme is depicted in Figure 9.11
- As we can see from this figure, each cluster has a CH which collects data from its cluster members, aggregates it and sends it to the BS or an upper level CH

# *Hierarchical Routing*







# *Cluster Based Routing Protocol*

## *(CBRP)*

- A simple cluster based routing protocol (CBRP) divides the network nodes into a number of overlapping or disjoint two-hop-diameter clusters in a distributed manner
- The cluster members just send the data to the CH, and the CH is responsible for routing the data to the destination
- The major drawback with CBRP is that it requires a lot of hello messages to form and maintain the clusters, and thus may not be suitable for WSN
- Given that sensor nodes are stationary in most of the applications this is a considerable and unnecessary overhead

### *Scalable Coordination*

- In hierarchical clustering method, the cluster formation appears to require considerable amount of energy as periodic advertisements are needed to form the hierarchy
- Also, any changes in the network conditions or sensor energy level result in re-clustering which may be not quite acceptable as some parameters tend to change dynamically



# *Low-Energy Adaptive Clustering Hierarchy*

---

## Low-Energy Adaptive Clustering Hierarchy (**LEACH**)

- LEACH is a hierarchical clustering algorithm for sensor networks, called Low-Energy Adaptive Clustering Hierarchy (LEACH)
- LEACH is a good approximation of a proactive network protocol, with some minor differences which includes a distributed cluster formation algorithm
- LEACH randomly selects a few sensor nodes as CHs and rotates this role amongst the cluster members so as to evenly distribute the energy dissipation across the cluster
- In LEACH, the CH nodes compress data arriving from nodes that belong to the respective cluster, and send an aggregated packet to the BS in order to reduce the amount of information that must be transmitted
- LEACH uses a TDMA and CDMA MAC to reduce intra-cluster and inter-cluster collisions, respectively, while data collection is centralized and is performed periodically





# *Power-Efficient Gathering in Sensor Information Systems (PEGASIS)*

- The Power-Efficient Gathering in Sensor Information Systems (**PEGASIS**) is a near optimal chain-based protocol which is an enhancement over LEACH
- In order to prolong network lifetime, nodes employing PEGASIS communicate with their closest neighbors only and they take turns in communicating with the BS
- Whenever a round of nodes communicating with the BS ends, a new round starts
- This decreases the power required to transmit data per round, as energy dissipation is spread uniformly over all nodes and as a result, PEGASIS has **two main goals**
- First, it aims at increasing the lifetime of each node by using collaborative techniques, as a result, the overall network lifetime is also increased
- Second, it only allows coordination between nodes that are close together, thus reducing the bandwidth consumed for communication
- To locate the closest neighbor SN, SNs use the signal strength to measure the distance to all of its neighboring nodes and then adjust the signal strength so that only one node can be heard





# *Small Minimum Energy Communication Network (MECN)*

- The minimum energy communication network (MECN) protocol has been designed to compute an energy-efficient subnetwork for a given sensor network
- On top of MECN, a new algorithm called Small MECN (SMECN) has been proposed to construct such a subnetwork
- The subnetwork (i.e. , subgraph  $G'$ ) constructed by SMECN is smaller than the one constructed by MECN if the broadcast region around the broadcasting node is circular for a given power assignment
- The subgraph  $G'$  of graph  $G$ , which represents the sensor network, minimizes the energy consumption satisfying the following conditions:
  - The number of edges in  $G'$  is less than in  $G$ , while containing all nodes in  $G$
  - The energy required to transmit data from a node to all its neighbors in subgraph  $G'$  is less than the energy required to transmit to all its neighbors in graph  $G$
- The resulting subnetwork computed by SMECN helps in the task of sending messages on minimum-energy paths



# *Threshold-sensitive Energy Efficient (TEEN)*

---

## *Threshold-sensitive Energy Efficient (TEEN)*

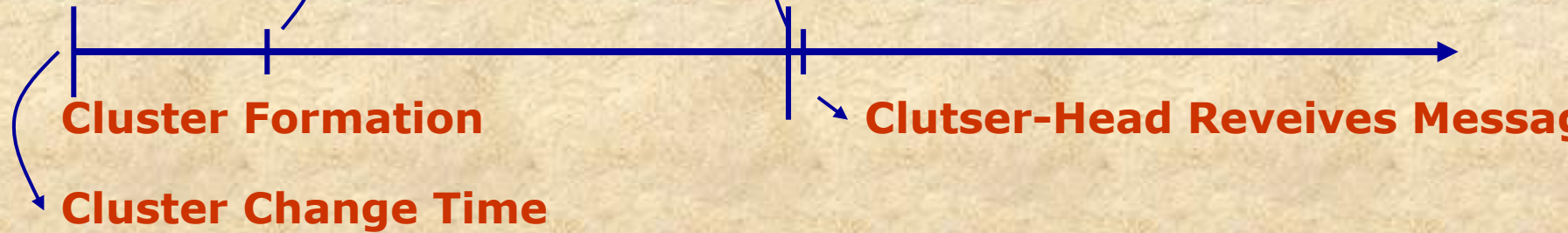
- A Threshold-sensitive Energy Efficient sensor Network (TEEN) protocol has been depicted in Figure 9.12
- In this scheme, at every cluster change time, the CH broadcasts the following to its members in addition to the attributes:
  - Hard Threshold (HT): This is a threshold value for the sensed attribute
  - It is the absolute value of the attribute beyond which, the node sensing this value must switch on its transmitter and report to its CH
  - Soft Threshold (ST): This is a small change in the value of the sensed attribute which triggers the node to switch on its transmitter and transmit once the HT has been crossed
- In TEEN, nodes sense their environment continuously, thereby making it appropriate for real time applications



# *Threshold-sensitive Energy Efficient (TEEN)*

Parameters

Attribute > Threshold



## ▪ Features

- ❑ Suited for time critical sensing applications
- ❑ Time critical data reaches the user almost instantaneously
- ❑ At every cluster change time, the parameters are broadcast afresh and so, the user can change them as required
- ❑ Energy consumption can be controlled by changing the threshold values





# *TEEN Reactive Protocol*

---

- **Features**

- It offers flexibility by allowing the user to set the threshold values for the attributes
- Attributes can be changed every cluster change

- **Drawback**

- If threshold  $H_T$  not reached then user never gets to know about the network

- **Application**

- Time critical environment like intrusion detection, etc.



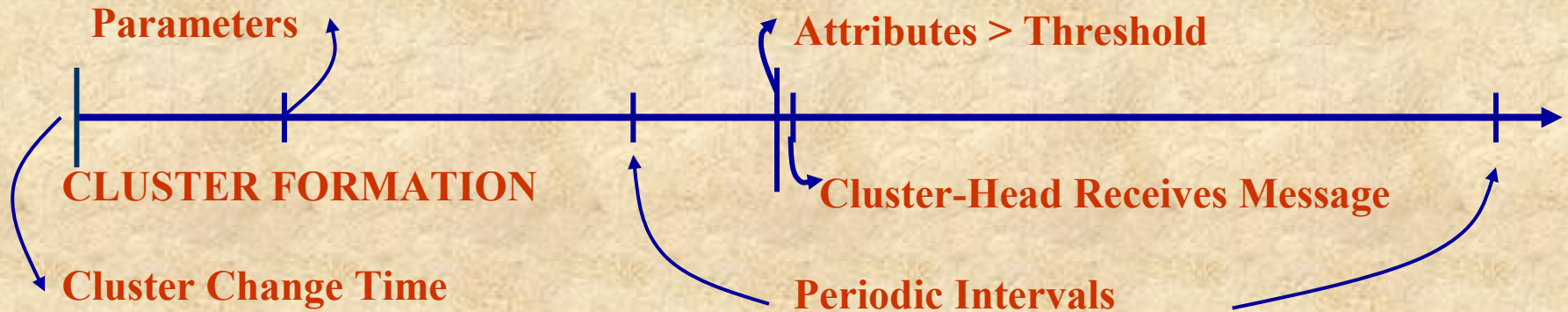
# *Hybrid Protocol (UC)*

---

- **APTEEN** (Adaptive Periodic Threshold Sensitive Energy Efficient Sensor Network Protocol)
  - ▣ Combines features of Reactive and Proactive Networks
  - ▣ Periodicity and threshold values controlled by the user
  - ▣ Energy Efficient Protocol

# Hybrid Protocol (APTEEN)

- To take advantage of both the networks, it is preferable to have both the features in the system (UC)



- **Functioning**
  - ▣ [Attributes, Thresholds ( $H_T, S_T$ ), Count Time ( $T_C$ )] are broadcast to all cluster members





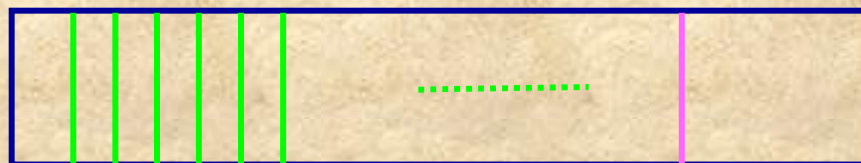
# *Hybrid Protocol*

---

- **Features:**
  - ❑ **Combines both proactive and reactive policies**
  - ❑ **Offers a lot of flexibility by allowing the user to set the time interval and the threshold values for the attributes**
  - ❑ **Energy consumption can be controlled by changing periodic interval as well as the threshold values**
  - ❑ **Can emulate either a proactive network or a reactive network, based on the application**
  - ❑ **Drawback: Increased complexity**

# *Modified TDMA for APTEEN*

- Time-critical queries and historical queries are answered by the BS
- Based on the assumption that adjacent nodes sense similar data, we can make only one of them handle the query
  - ▣ This might reduce the accuracy of data for non-critical queries
  - ▣ This is acceptable since it almost doubles the life of the network



Original TDMA



Modified TDMA





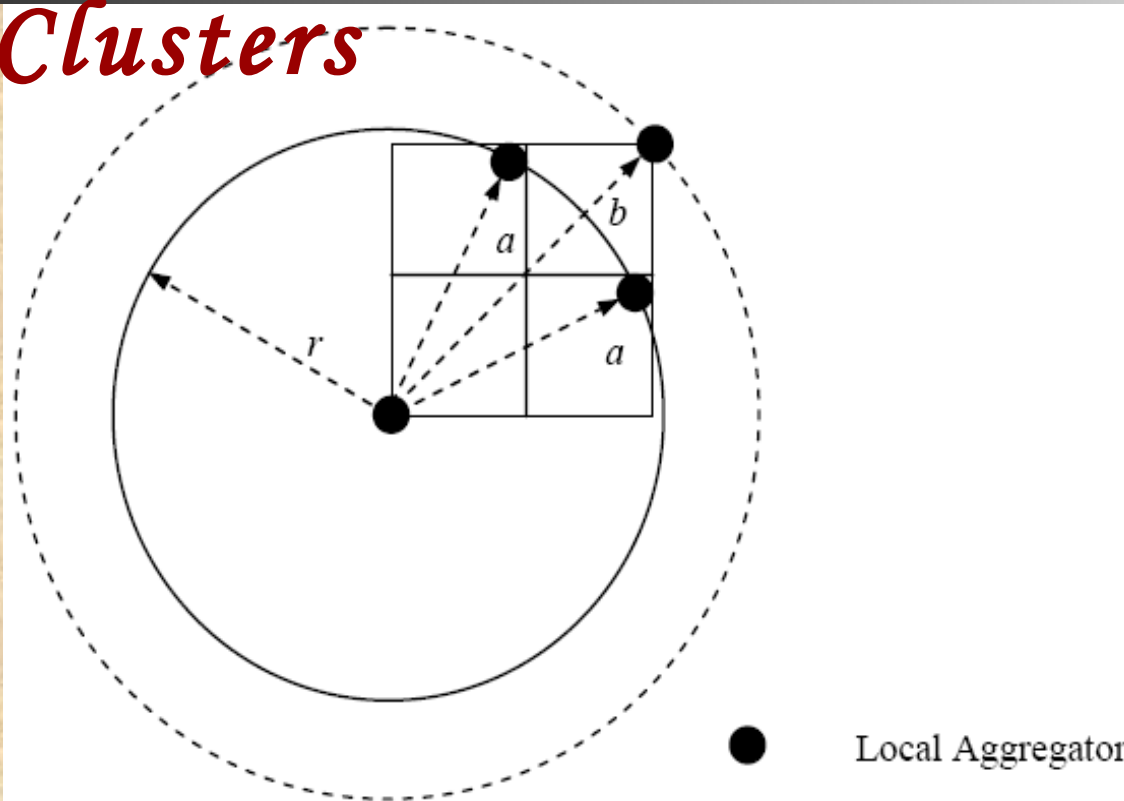
# *Routing in Fixed-size Clusters*

---

- Routing in sensor networks can also take advantage of geography-awareness
- One such routing protocol is called Geography Adaptive Fidelity (GAF) where the network is firstly divided into fixed zones
- Within each zone, nodes collaborate with each other to play different roles
- For example, nodes elect one SN to stay awake for a certain period of time while the others sleep
- This particular elected SN is responsible for monitoring and reporting data to the BS on behalf of all nodes within the zone
- Here, each SN is positioned randomly in a two dimensional plane
- When a sensor transmits a packet for a total distance  $r$ , the signal is strong enough for other sensors to hear it within the Euclidean distance  $r$  from the sensor that originates the packet
- Figure 9.14 depicts an example of fixed zoning that can be applied to WSN



# Routing in Fixed-size Clusters



**$r$  is the distance of packet transmission by each sensor**

- A fixed cluster of each side  $a$  can be selected and is connected if:
  - ▣ If the signal travels a distance of  $a = r/(\sqrt{5})$  in adjacent vertical or horizontal directions, two sensors can communicate directly
  - ▣ For a diagonal communication to take place, the signal has to span a distance of  $a = r/(2\sqrt{2})$



# *Sensor Aggregates Routing*

---

- A sensor aggregate includes those SNs in a network that satisfy a grouping predicate for a collaborative processing task
- The parameters of the predicate depend on the task and its resource requirements
- Here, the formation of appropriate sensor aggregate is considered in terms of resource allocation for communication and sensing
- Sensors in the network are divided into clusters according to their sensed signal strength
- After that, local cluster leaders (CH) are elected by exchanging information between neighboring sensors
- Once a sensor node has exchanged packets with all its one-hop neighbors, if it finds that its signal strength is higher than all its one-hop neighbors, it declares itself as a leader
- This leader-based tracking algorithm assumes a unique leader to know surrounding geographical region for collaboration



# *Hierarchical Power-Aware Routing*

---

- A hierarchical power-aware routing scheme divides the network into groups of sensors
- The groups in a geographic proximity, are clustered together as a zone and each zone is treated as an entity
- Routing is performed by allowing each zone to decide how it routes a message hierarchically across other zones
- In this scheme, messages are routed along the path with the maximal fraction of the remaining power after the message is transmitted, and this is called the max-min path
- One of the concerns with the max-min path is that traversal through the SNs with high residual power may be expensive as compared to the path with the minimal power consumption
- Too much power consumption decreases the overall power level of the system, thereby decreasing the lifetime of the network





# *Sensor Protocols for Information via Negotiation (SPIN)*

- Disseminates all the information of each SN to every other SN in the network
- All SNs in the network are potential BS
- A user is able to query any SN and get the required information immediately
- These protocols make use of the property that SNs in close proximity have similar data and thus transmit only the data that the other SNs do not have
- SPIN assigns a high-level name to appropriately describe their collected data, called meta-data, and perform meta-data negotiations before any data is transmitted
- This ensures that no redundant data is transmitted throughout the network
- The format of the meta-data is application-specific and is not specified in SPIN
- SPIN works in a time-driven manner wherein it distributes the information all over the network, even when a user does not request any data
- The SPIN family of protocols includes two protocols, namely, SPIN-1 and SPIN-2, which incorporate negotiation before transmitting data so as to ensure that only useful information is transferred



# *Flat versus Hierarchical*

---

## **Hierarchical**

**Reservation-based scheduling**

**Collisions avoided**

**Reduced duty cycle due to periodic sleeping**

**Data aggregation by cluster head**

**Simple but non-optimal routing**

**Requires global and local synchronization**

**Overhead of cluster formation throughout the network**

**Lower latency as multi-hop network formed by cluster heads is always available**

**Energy dissipation is uniform**

**Energy dissipation cannot be controlled**

## **Flat**

**Contention-based scheduling**

**Collision overhead present**

**Variable duty cycle by controlling sleep time of nodes**

**Node on multi-hop path aggregates incoming data from neighbors**

**Routing is complex but optimal**

**Links formed in the fly, without synchronization**

**Routes formed only in regions that have data for transmission**

**Latency in waking up intermediate nodes and setting up the multi-hop path**

**Energy dissipation depends on traffic patterns**

**Energy dissipation adapts to traffic pattern**

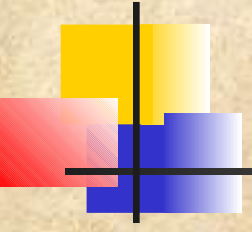


# *Comparison of SPIN, LEACH and Directed Diffusion*

---

<b>Protocol →</b>	<b>SPIN</b>	<b>LEACH</b>	<b>Directed Diffusion</b>
<b>Optimal Route</b>	<b>No</b>	<b>No</b>	<b>Yes</b>
<b>Network Lifetime</b>	<b>Good</b>	<b>Very Good</b>	<b>Good</b>
<b>Resource Awareness</b>	<b>Yes</b>	<b>Yes</b>	<b>Yes</b>
<b>Use of Meta-Data</b>	<b>Yes</b>	<b>Yes</b>	<b>Yes</b>





# *Negotiation-based Routing*

---

- **Negotiation-based routing protocols use high level data descriptors in order to eliminate redundant data transmissions**
- **Communication decisions are also made based on the available resources**
- **The motivation here is that the use of flooding to disseminate data produces implosion and data overlap, leading to scenarios where nodes receive duplicate copies of the same data**
- **If the same data is transmitted by several sensors, considerable energy is consumed**
- **The main idea behind negotiation-based routing in WSNs is to suppress duplicate information and prevent redundant data from being sent to the next sensor or the BS**
- **This is done by conducting a series of negotiation messages before the actual data transmission begins**



# *Multipath-based Routing*

---

- Network performance, and possibly lifetime, in WSNs can be significantly improved if the routing protocol is able to maintain multiple paths to a destination
- The fault tolerance (resilience) is considerably increased, which is measured by the likelihood that an alternate path exists between a source and a destination when the primary path fails
- Clearly, this can be increased if we maintain multiple paths between the source and the destination at the expense of an increased energy consumption and traffic generation (i.e., overhead), as alternate paths are kept alive by sending periodic messages
- We would also like to note here that multipath routes between a source and a destination can be node-disjoint or not
- Multiple paths between a source and destination are said to be node-disjoint when there is no node overlap amongst them





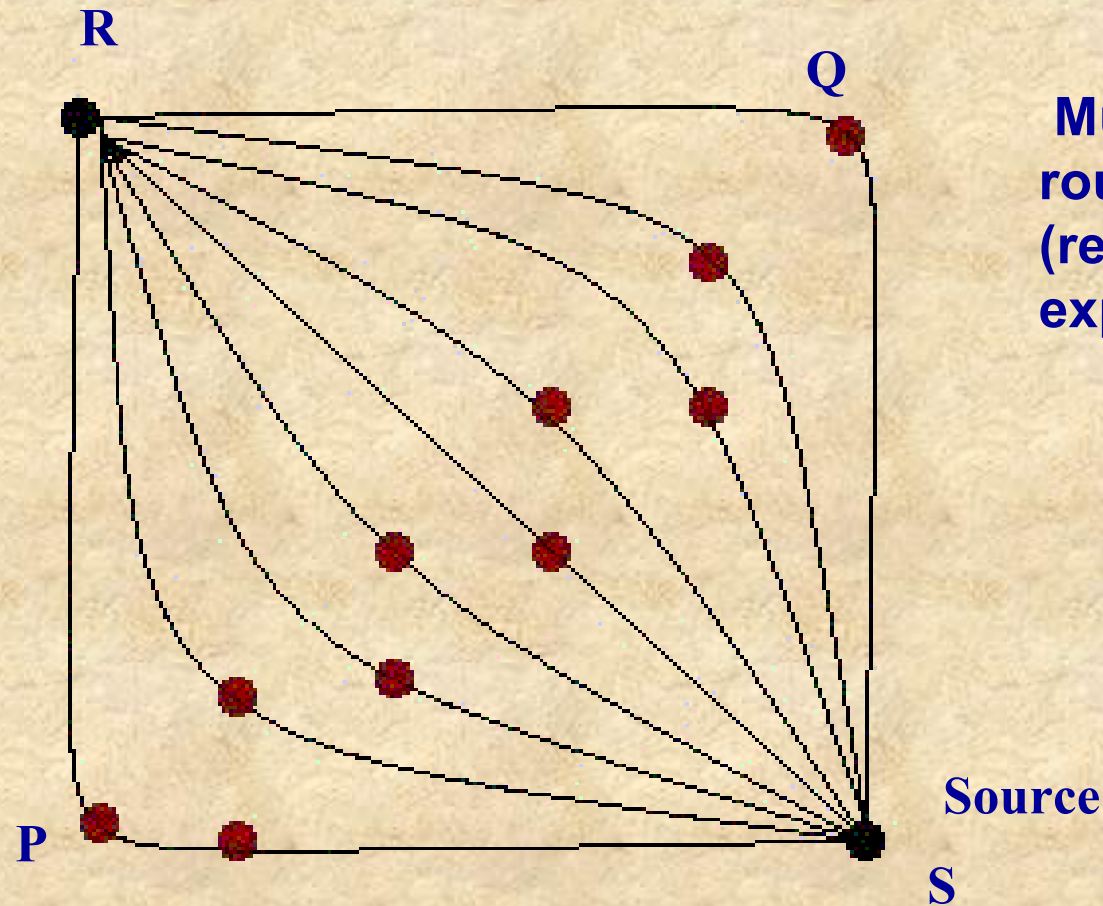
# *Multipath-based Routing*

---

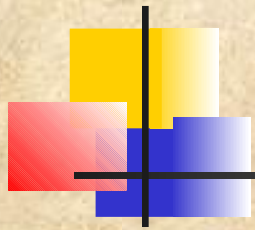
- An extension of the multi-path algorithm is described that contains several important characteristics
- The idea is to reduce the complexity of finding the paths by defining the rectangular region bounded by the responding sensor and the BS as the routing region and defining the paths passing through cross-diagonal sensors as multiple paths
- One such example for a rectangular mesh-based WSN, is shown in Figure 9.15
- This identifies many paths, with different path lengths in terms of number of intermediate SNs in the path and hence, reduce the delay between the responding SN and the BS by the process of data store-and-forward along the selected path
- The path along the diagonal, is shortest in length and if this path is used all the time in responding the persistent query, the energy of the sensors lying on this path, could get depleted at a much faster rate than rest of the network



# *Multipath-based Routing*

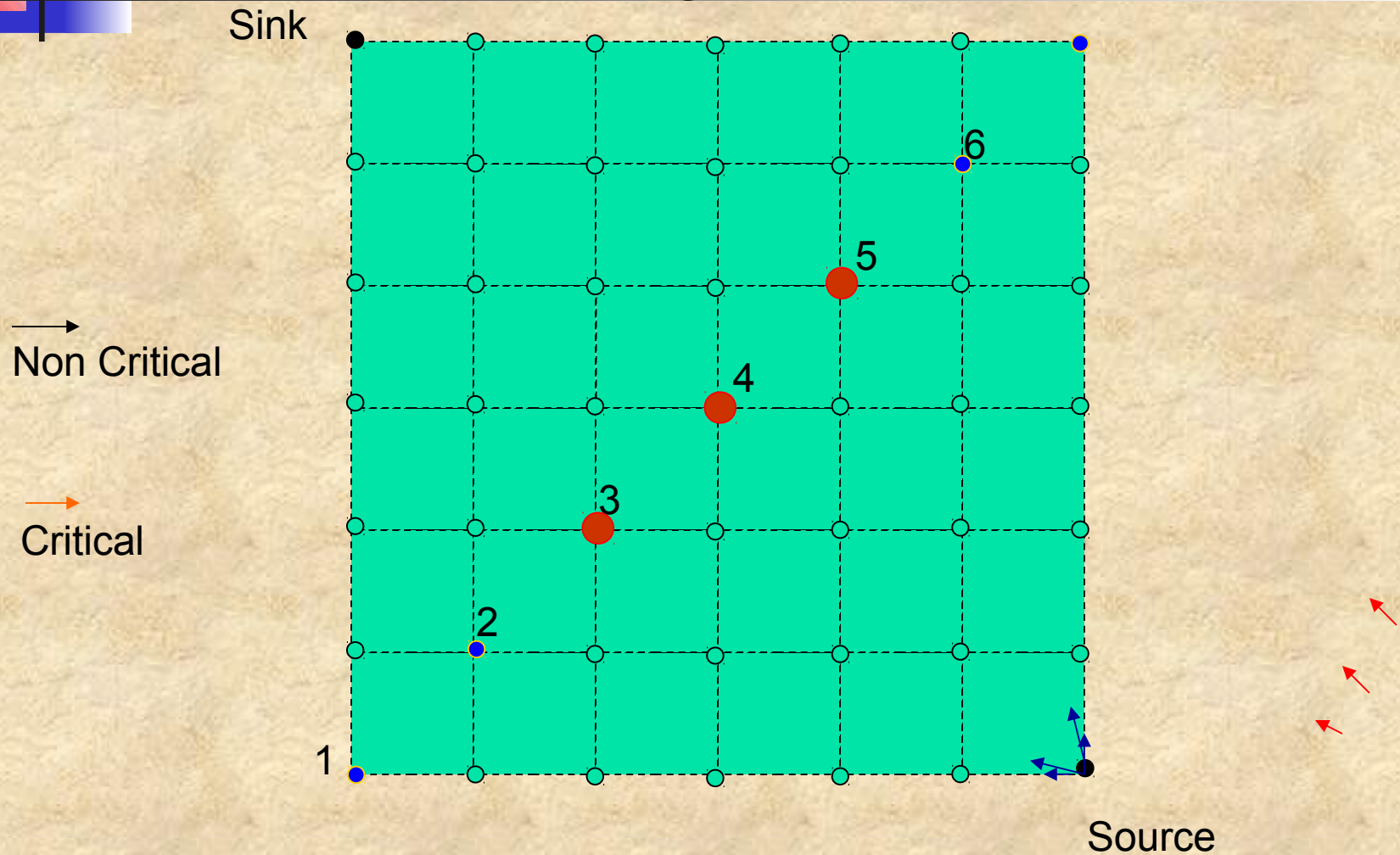


**Multiple Paths are used to route query responses (resemble sides of an expanding rhombus)**



# *Load Shedding*

7





# *Query-based Routing*

---

- In query-based routing, the destination nodes propagate a query for data (sensing task) from a node throughout the network
- A node having the data matching the query sends it back to the node which requested it
- Usually, these queries are described in natural language or in high-level query languages
- For example, a BS B1 may submit a query to node N1 inquiring: “Are there moving vehicles in battlefield region 1?”
- In query-based routing, all the nodes have tables consisting of the sensing tasks queries that they received, and send back data matching these tasks whenever they receive it
- Directed diffusion (discussed earlier in this chapter) is an example of this type of routing
- Here, the sink node sends out messages of interest to SNs
- As the interest is propagated throughout the WSN, the gradients from the source back to the sink (BS) are set up





# *Location-based Routing*

---

- In location-based routing, SNs are addressed by means of their locations
- Here, the distance between neighboring SNs can be estimated on the basis of incoming signal strengths, and relative coordinates of neighboring SNs can be obtained by exchanging such information
- Alternatively, the location of nodes may be available directly through GPS if we consider nodes are equipped with a small low power GPS receiver
- In order to conserve energy, some location-based schemes demand that SNs should go to sleep if there is no activity
- Clearly, the more sleeping SNs in the network the more energy can be saved
- However, the active SNs should be connected, should cover the entire sensing region, and should provide basic routing and broadcasting functionalities



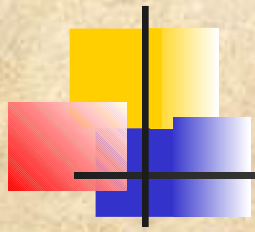
# *High-Level Application Layer Support*

---

- The protocols we have presented so far are also found, albeit in some different form in traditional wired, cellular, or ad hoc networks
- For specific applications, a higher level of abstraction specifically tailored to WSN appears to be useful

## **Distributed Query Processing**

- The number of messages generated in distributed query processing is several magnitudes less than in centralized scheme
- There are two approaches for processing sensor queries: warehousing and distributed
- In the warehousing approach, data is extracted in a pre-defined manner and stored in a central database
- In the distributed approach, only relevant data is extracted from the sensor network, when and where it is needed



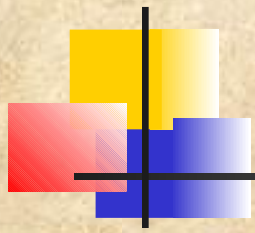
# *High-Level Application Layer*

## *Support*

### **Sensor Databases**

- One can view the wireless sensor network as a comprehensive distributed database and interact with it via database queries
- This approach solves, en passant, the entire problem of service definition and interfaces to WSNs by mandating, for example, SQL queries as the interface
- The problems encountered here are in finding energy-efficiency ways of executing such queries and of defining proper query languages that can express the full richness of WSNs
- The TinyDB project carried out at the University of California at Berkeley is looking at these issues
- A model for sensor database systems known as COUGAR, defines appropriate user and internal representation of queries
- The sensor queries is also considered so that it is easier to aggregate the data and to combine two or more queries





# *High-Level Application Layer*

## *Support*

### **Distributed Algorithms**

- WSNs are not only concerned with merely *sensing* the environment but also with interacting with the environment
- Once actuators like valves are added to WSNs, the question of distributed algorithms becomes inevitable
- One showcase is the question of distributed consensus, where several actuators have to reach a joint decision (a functionality which is also required for distributed software update, for example)

### **In-Network Processing**

- In-network processing, requires data to be modified as it flows through the network
- It has become one of the primary enabling technologies for WSNs as it has the potential to considerably increase the energy efficiency of the network



# *Security*

---

- **Security for wireless sensor networks is still a wide open field**
- **Much work seems to be directly transferred from the MANET case, but the principal threats and possible attacks to the correct functioning of WSNs are still missing a thorough analysis (albeit they will most certainly be largely application-dependent)**
- **In any case, in the next chapter we present the existing security solutions in the context of not only ad hoc networks, but also wireless sensor networks**
- **However, we note that there is still much to do and this is a wide open field for research**



# *Adapting to the Inherent Dynamic Nature of WSNs*

---

Some important goals that current research in this area is aiming to achieve are as follows:

- ▣ Exploit spatial diversity and density of sensor/actuator nodes to build an adaptive node sleep schedule
- ▣ Spontaneously create and assemble network, dynamically adapt to device failure and degradation, manage mobility of sensor nodes and react to changes in task and sensor requirements
- ▣ Adaptability to drastic changes in the traffic
- ▣ Having finer control over the precision and coverage
- The Scalable Coordination Architectures for Deeply Distributed Systems (SCADDS) project, also a part of DARPA SensIT program, focuses on adaptive fidelity, dynamically adjusting the overall fidelity of sensing in response to task dynamics (turn on more sensors when a threat is perceived)





# *Conclusions and Future Directions*

---

- WSNs are perhaps one of the fastest growing areas in the broad wireless ad hoc networking field
- The research in WSNs is flourishing at a rapid pace and is being considered as the revolutionary concept of this century
- But, there are many challenges that need to be addressed such as, how to miniaturize the power source, how to have a self-power generating technology to provide indefinite power source and how to provide secured communication without exceeding the resource requirements
- Another area that needs serious investigation is to come up with a **killer non-defense civilian application** so as to enhance its usefulness and general acceptance
- The challenges are many and we have partial answers or roadmaps to some of the above questions, there is still much to be done